

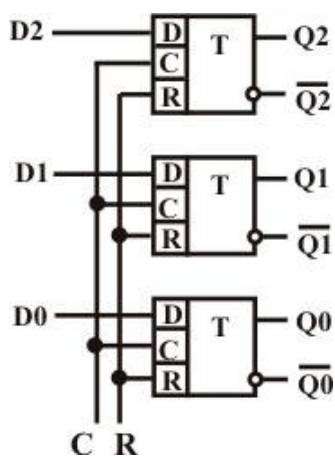
Урок 5. Регистры, счетчики

В предыдущем уроке были рассмотрены основные виды триггеров. Напомним, что все виды триггеров основаны на использовании базового асинхронного RS-триггера.

Регистр хранения

На триггерах можно строить более сложные цифровые устройства, например такие, как регистры. Регистры предназначены для хранения многобитовой информации, то есть чисел, записанных в двоичном коде.

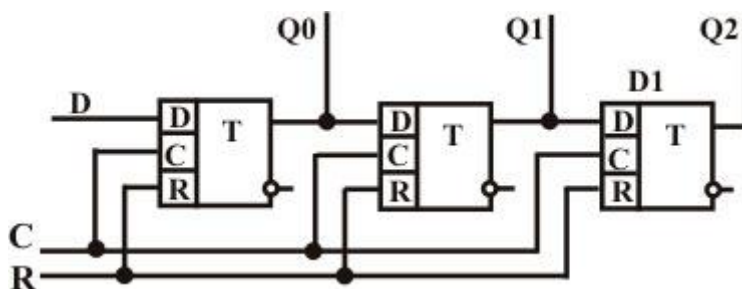
Рассмотрим трех битовый регистр хранения на D-триггерах:



Каждый триггер может хранить один разряд (бит) числа. Вход R служит для установки выходов всех триггеров в нулевое (исходное) состояние перед записью числа, которое подается на входы D0, D1 и D2. При подаче импульса на вход C производится запись информации с этих входов. Информация может храниться сколь угодно долго, пока на вход C не подаются импульсы и подается питание.

Регистр сдвига

Другой разновидностью регистров является регистр сдвига. Он предназначен для преобразования информации путем ее побитного сдвига в ту или иную сторону. На следующем рисунке приведена схема простейшего регистра сдвига информации вправо (по схеме):



В отличие от регистра хранения выход предыдущего триггера соединен с входом последующего. Информация в виде логического уровня подается на вход первого (крайнего слева) триггера. При воздействии импульса на входе C присутствующая на входе D информация записывается в первый триггер. При подаче второго импульса информация из первого триггера переписывается во второй триггер, а в первый

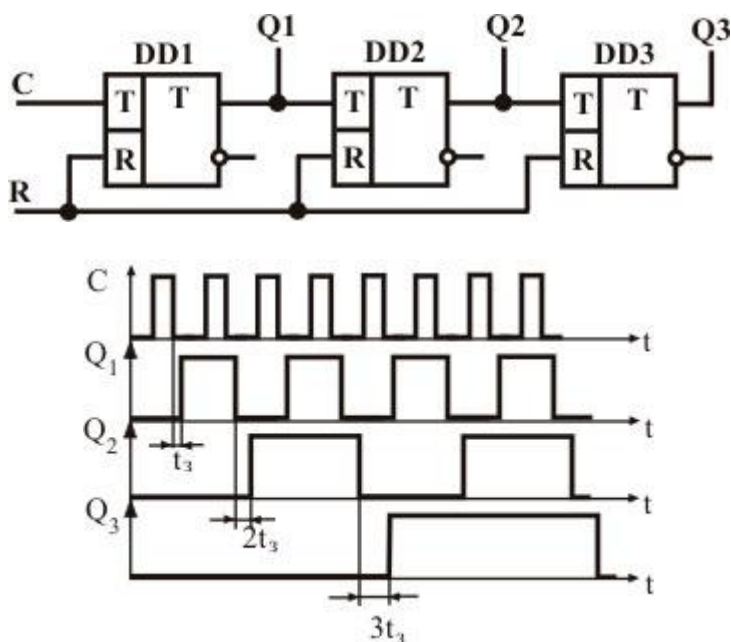
записывается информация, которая в этот момент присутствует на входе D, и так далее. Таким образом, с подачей каждого синхроимпульса информация в регистре сдвигается вправо на 1 разряд.

Сдвиговые регистры используются во многих схемотехнических решениях при построении цифровых устройств, прежде всего для преобразования последовательного кода в параллельный, а также для выполнения арифметических операций (умножения и деления на 2) с двоичными числами, организации линий задержки, формирования импульсов заданной длительности, генерирования псевдослучайных последовательностей (кодов) и т.п.

Счетчик

Еще один класс цифровых устройств, которые можно построить на триггерах – счетчики. Как следует из названия, они осуществляют счет входных импульсов в заданном коде и могут хранить результат.

Простейший счетчик с последовательным переносом можно получить с помощью счетных Т-триггеров:



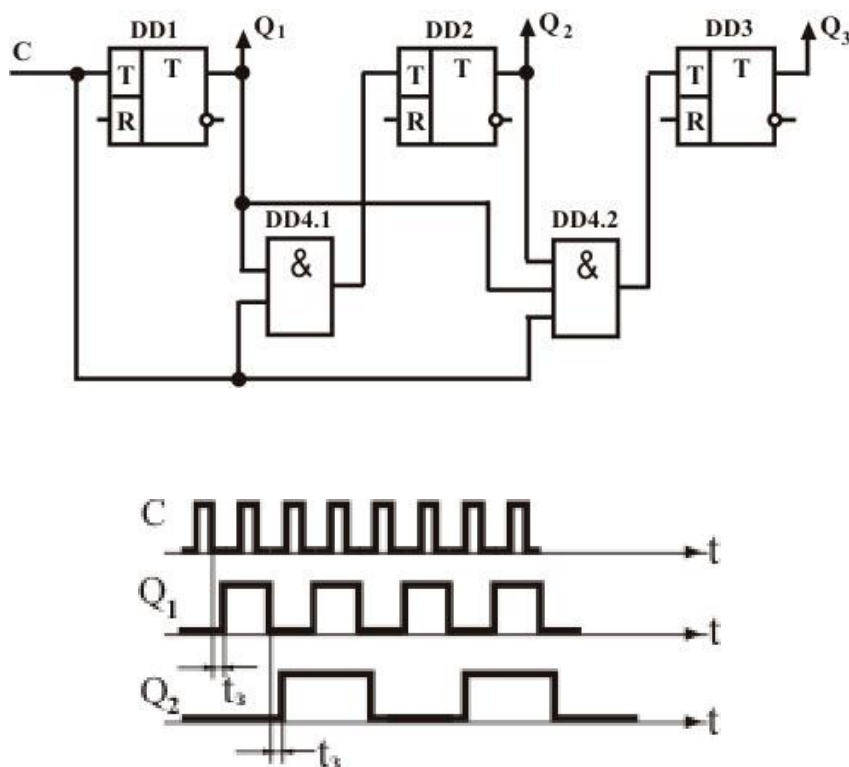
Подачей импульса на вход R счетчик приводится в исходное состояние, когда на выходах Q1-Q3 – уровень логического нуля.

На вход C подаются импульсы для счета. С приходом заднего фронта первого импульса первый (левый) по схеме триггер устанавливается в «1». Если читать код справа налево, то он соответствует единице. Для нашего трехразрядного счетчика это код 001. С приходом второго импульса в «1» переключается второй триггер, а первый переключается в «0». Таким образом, код на выходах счетчика будет 010, что соответствует десятичной цифре 2. Следующий импульс установит код 011, то есть 3. Трехразрядный счетчик может досчитать до кода 111, что соответствует десятичной цифре 7. При этом наступает так называемое переполнение счетчика, и с приходом следующего импульса счетчик обнулится.

Поскольку триггеры счетчика соединены последовательно, то и переключаться они будут также последовательно. Этот процесс отображен на графике, из которого видно, что

время задержки переключения t_z будет удваиваться и утраиваться. С увеличением числа разрядов задержка может оказаться неприемлемой, что является недостатком счетчиков с последовательным переносом.

Для повышения быстродействия применяются счетчики с параллельным переносом, что достигается одновременной подачей входных импульсов на входы всех триггеров счетчика. Это реализуется с помощью введения в схему логических элементов И:



Из схемы видно, что на вход второго триггера счетный импульс поступит только тогда, когда на выходе первого триггера будет «1», а на третий – когда «1» будет на выходах и первого, и второго триггеров. Очевидно, что с увеличением числа разрядов необходимо увеличивать как число логических элементов И, так и число их входов, что, в свою очередь, является недостатком такого типа счетчиков.

Регистры и счетчики, в свою очередь, могут применяться для построения более сложных цифровых устройств: сумматоров, ОЗУ и ПЗУ (оперативных и постоянных запоминающих устройств), АЛУ (арифметическо-логических устройств), входящих в состав процессоров, и так далее, к все более сложным цифровым устройствам.